

9

Le Product Owner

Dans ce chapitre, je décris en détail le rôle de Product Owner. J'explique d'abord l'utilité de ce rôle en relation avec les autres rôles Scrum, puis je rappelle les principales responsabilités et les principaux attributs de ce rôle. Je vous propose ensuite de vivre un exemple du quotidien d'un Product Owner sur plusieurs semaines. J'indique ensuite qui est le plus apte à endosser le rôle de Product Owner selon le type de développement et je conclus en montrant que ce rôle peut être combiné à d'autres ou, inversement, qu'il peut être réparti entre plusieurs personnes.

Présentation

Le Product Owner est le point d'articulation et d'autorité qui dirige le développement d'un produit. C'est l'un des trois rôles coopératifs de toute équipe Scrum (les deux autres étant le ScrumMaster et l'équipe de développement).

Le Product Owner se singularise par son besoin de porter son attention dans deux directions à la fois (voir Figure 9.1).

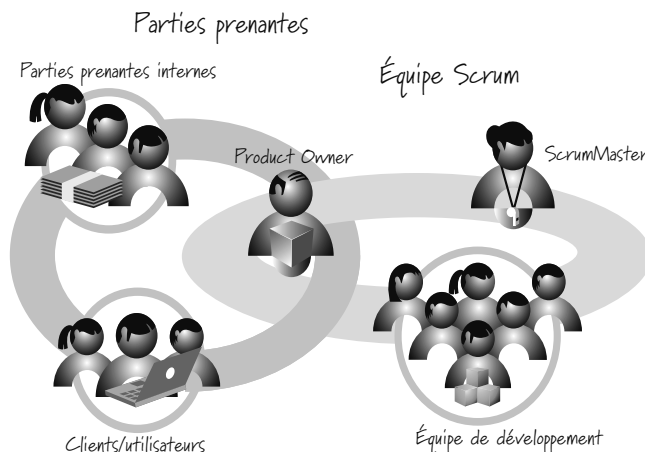


Figure 9.1

Le Product Owner se concentre dans deux directions simultanément.

D'un côté, le Product Owner doit bien comprendre les besoins et les priorités des parties prenantes internes et externes (direction, clients, utilisateurs). Il doit en effet agir en leur nom. Le Product Owner est donc un product manager qui garantit que la solution qui sera développée sera la bonne.

D'un autre côté, le Product Owner doit rester en communication permanente avec l'équipe de développement au sujet de ce qu'il faut produire et dans quel ordre. Il doit s'assurer que les critères de validation/recette sont spécifiés et que les tests qui permettront de vérifier et d'accepter le produit seront bien exécutés dès que les fonctions seront produites. Le Product Owner ne va pas définir en détail ces tests, mais il vérifie que les tests de haut niveau ont été rédigés. Cela permettra à l'équipe de savoir quand elle peut déclarer que la fonction est achevée et satisfera le Product Owner. Sous cet aspect, le Product Owner est également en partie un analyste métier et un testeur.

Responsabilités principales

La Figure 9.2 schématise les principales responsabilités du Product Owner.

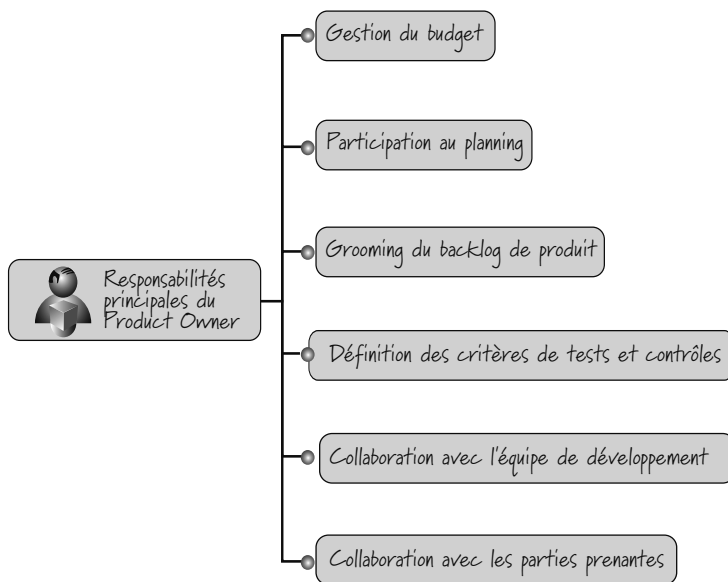


Figure 9.2

Principales responsabilités du Product Owner.

Il s'agit à l'évidence d'un rôle à temps complet avec des responsabilités majeures. À la lecture des descriptions qui suivent, vous pourriez penser que détenir ces responsabilités est trop pour une seule personne, ou qu'il sera difficile de trouver un profil possédant tous les attributs requis pour réussir. Pourtant, une même personne parvient en général à endosser ce rôle. Dans certaines circonstances rares, il faudra former une équipe de Product Owners ou désigner des Product Owners délégués. J'y reviendrai en fin de chapitre.

Gestion du budget

Le Product Owner doit s'assurer que des décisions budgétaires pertinentes sont prises en permanence au niveau d'une release, du Backlog de produit et de chaque sprint (voir Figure 9.3).

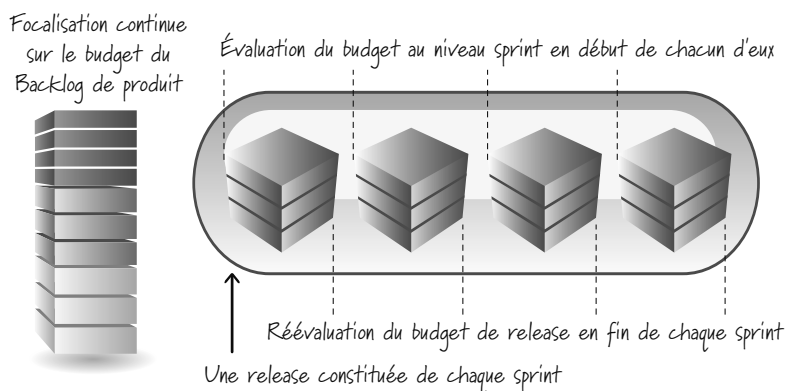


Figure 9.3

Gestion budgétaire par le Product Owner.

Budget au niveau release

Au niveau d'une release, le Product Owner effectue sans cesse des arbitrages en termes de périmètre fonctionnel, de date de livraison, de budget et de qualité, afin de tirer profit du flux d'informations significatives au niveau budgétaire qui arrive au cours du développement. Les arbitrages effectués en début de cycle peuvent ne plus s'avérer pertinents suite à l'arrivée d'une nouvelle information pendant la release.

Prenons un exemple : nous avons déjà progressé de plusieurs semaines dans un effort de développement à date fixe sur six mois. Nous constatons alors qu'est apparue une opportunité d'augmenter le chiffre d'affaires de 50 % si nous acceptons de consacrer une semaine de plus (4 % de décalage) pour ajouter une nouvelle fonction à cette release. Devons-nous repousser la sortie d'une semaine et supporter le coût correspondant pour augmenter le CA ? C'est au Product Owner de trancher, ce qu'il peut faire en général de façon unilatérale. Parfois, il peut recommander une décision tout en travaillant avec les autres personnes pour s'assurer de leur soutien dans l'exécution (et parfois de leur approbation).

En fin de sprint, le Product Owner dirige la prise de décision pour financer ou non le sprint suivant. La réponse sera positive s'il constate une progression correcte en direction du but de la release ou si le sprint suivant possède une autre justification économique. Si la progression est insuffisante ou si le budget ne suffit plus, l'effort prévu peut être annulé.

Un Product Owner peut également décider de stopper le financement des développements ultérieurs à la fin d'un sprint s'il est satisfait du produit et qu'il le juge prêt à être livré, les compléments ne se justifiant plus. Supposons par exemple que nous ayons prévu une release sur dix sprints. Au bout du sprint 7, le Product Owner passe en revue

les éléments restants dans le Backlog de produit et en conclut que le coût de production de ces éléments est supérieur au complément de valeur qu'ils vont rapporter. Il peut donc décider de lancer le produit au lieu de laisser se poursuivre le plan d'origine en dix sprints. Cette souplesse permettant d'écourter un développement n'est possible que si on a pris soin de faire produire les éléments apportant le plus de valeur en premier, en les plaçant en haut du Backlog de produit. Cela suppose également que l'équipe ait vraiment achevé le travail lors de chaque sprint en conformité avec une définition d'achèvement rigoureuse.

Le Product Owner peut également décider d'arrêter le financement à la fin d'un sprint du fait que le profil financier fondamental a évolué. Citons l'exemple de la création d'un produit spécifique à un pays. Un beau jour, le gouvernement de ce pays vote une loi qui ôte tout espoir de rentabilité au produit ou même le rend illégal. Dans une telle situation, le Product Owner peut envisager l'arrêt immédiat de l'effort de développement, même si tout se déroulait correctement.

Budget au niveau sprint

Le Product Owner est également en charge de la gestion budgétaire au niveau des sprints, car il doit s'assurer d'un bon retour sur investissement (ROI) pour chaque sprint. Un Product Owner sérieux gère les finances de son entreprise comme si c'était son propre argent. Normalement, il connaît le coût du prochain sprint (puisque la durée et la composition de l'équipe ont été définies). Il peut donc s'interroger pendant la planification du sprint : "Suis-je prêt à faire un chèque personnel du montant de ce sprint pour obtenir les fonctions que nous avons prévu de produire ?" En cas de réponse négative, le Product Owner sérieux ne dépensera pas non plus l'argent de son entreprise.

Budget au niveau Backlog de produit

J'ai indiqué dans le Chapitre 6 que le Product Owner devait créer des priorités entre les éléments du Backlog de produit (les PBI). Ces priorités doivent pouvoir évoluer lorsque les conditions budgétaires changent.

Imaginons par exemple qu'en début de release, le Product Owner considère une fonction comme très attrayante pour un fort pourcentage des utilisateurs visés. Parallèlement, l'équipe a considéré que sa production demanderait peu d'efforts. Au bout de quelques sprints, l'équipe change d'avis en constatant que l'effort pour produire la fonction sera bien supérieur. Par un malheureux concours de circonstances, il s'avère que cette fonction va intéresser bien moins d'utilisateurs que prévu. Le rapport coût-bénéfice de la fonction a donc énormément changé. Le Product Owner doit la repousser dans le Backlog de produit, en allant éventuellement jusqu'à supprimer les éléments PBI associés à cette fonction.

Participation au planning

Le Product Owner (PO) est un participant essentiel aux activités de planification des quatre niveaux : portefeuille, produit, release et sprint. Pendant la planification du portefeuille (Chapitre 16), le PO travaille avec les parties prenantes internes (par exemple, le

comité de direction ou le comité de pilotage) en sorte de bien positionner le produit dans le Backlog de portefeuille et décider quand commencer et terminer le développement.

- Pendant la planification de produit (Chapitre 17), le PO travaille avec les mêmes parties prenantes pour créer la vision globale du produit.
- Pendant la planification de release (Chapitre 18), le PO travaille avec les parties prenantes et avec l'équipe pour définir le contenu de la release suivante.
- Enfin, pendant la planification de chaque sprint (Chapitre 19), le PO travaille avec l'équipe de développement pour définir le but du sprint. Il fournit à cette équipe des données utiles pour lui permettre de sélectionner un jeu de PBI que l'équipe est assurée de pouvoir produire pour la fin du sprint.

Grooming du backlog de produit

Le Product Owner supervise l'activité préparatrice de Grooming, c'est-à-dire la création puis le peaufinage, l'estimation et l'ordonnancement (prioritisation) des éléments du Backlog de produit ou PBI (revoir le Chapitre 6). Il n'effectue pas lui-même tout le travail de Grooming. Ce n'est par exemple pas lui qui rédige la description des PBI, d'autres pouvant y contribuer. Ce n'est pas non plus le PO qui estime les éléments (c'est de la responsabilité de l'équipe de développement), mais il reste disponible pour répondre aux questions et clarifier pendant l'estimation. Le PO reste en revanche responsable en s'assurant que les activités de Grooming se déroulent de manière à favoriser une livraison de valeur continue.

Définition des critères de tests et contrôles

Le Product Owner doit définir les critères d'acceptation de chaque PBI. Ces critères correspondent aux conditions qui vont satisfaire le PO en prouvant que les spécifications fonctionnelles et non fonctionnelles sont satisfaites. Le PO peut rédiger des tests d'acceptation en fonction de ces critères ou bien demander le soutien d'experts métier (SME) ou de certains membres de l'équipe de développement. Dans tous les cas, le PO doit s'assurer que les critères d'acceptation (et souvent des tests spécifiques) sont définis avant qu'un élément puisse être discuté lors d'une réunion de planification de sprints. Si l'équipe ne dispose pas de ces tests, elle ne peut pas prétendre avoir une compréhension complète de l'élément à traiter et ne peut donc pas l'inclure dans un sprint. C'est pour cette raison que de nombreuses équipes Scrum requièrent la présence obligatoire de critères d'acceptation clairs dans leur liste de contrôle de définition d'achèvement (revoir le Chapitre 6).

C'est le Product Owner qui a la responsabilité en dernier ressort de contrôler que les critères d'acceptation ont été satisfaits. Il peut choisir de les vérifier lui-même ou bien demander le soutien d'utilisateurs experts afin de confirmer que les PBI produits remplissent les conditions de satisfaction. L'équipe peut aider à créer une infrastructure de test afin de permettre au PO ou à ses experts d'exécuter les tests plus efficacement. En fin de compte, c'est le PO qui décide si un élément satisfait aux exigences ou pas.

Il est important que le Product Owner contrôle les critères d'acceptation pendant l'exécution du sprint, sans attendre la réunion de revue de sprint. En réalisant quelques tests au fur et à mesure que les fonctions sont achevées, le PO peut plus rapidement vérifier les erreurs et les incompréhensions, ce qui laisse à l'équipe la possibilité de les corriger avant la revue de sprint. Par ailleurs, l'équipe n'est autorisée à présenter que des fonctions totalement achevées lors de la réunion de revue. Le PO doit donc s'assurer que les tests d'acceptation sont exécutés avant celle-ci pour que l'équipe sache quelles fonctions satisfont la liste de définition d'achèvement.

Collaboration avec l'équipe de développement

Le Product Owner doit bien sûr collaborer intimement et fréquemment avec l'équipe de développement. Il s'agit d'un rôle quotidien à fort engagement. De nombreuses entreprises qui commencent à adopter Scrum oublient de mettre l'accent sur l'engagement total du PO par rapport à l'équipe de développement. Cela entraîne un retard dans l'établissement de ce feedback si précieux et réduit significativement la valeur de ce feedback lorsqu'il finit par s'établir.

Ce manque d'engagement concerne également les personnes qui viennent d'être nommées PO et supposent que leur niveau d'investissement en approche Scrum doit ressembler à celui dont elles ont l'habitude en développement traditionnel. La Figure 9.4 compare le niveau d'engagement typique du client ou de la direction dans l'approche séquentielle traditionnelle et celle attendue d'un PO utilisant Scrum.

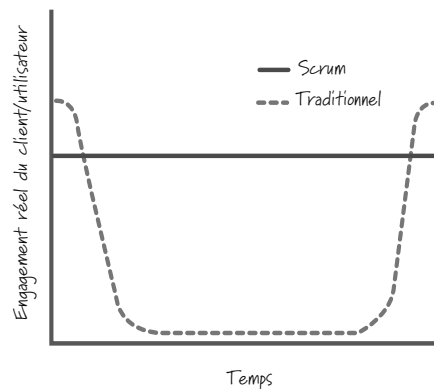


Figure 9.4

Comparaison de l'engagement du client ou de la direction en approche traditionnelle et Scrum.

Dans l'approche traditionnelle par phases, l'engagement forme un U. Les clients doivent s'investir beaucoup au départ pour bien définir les spécifications détaillées. Dès que l'on bascule dans les phases de production technique (conception, codage et certains tests), le client "n'est plus requis". Son niveau d'engagement reste assez bas ou nul pendant l'essentiel de l'effort. À vrai dire, dans l'approche traditionnelle, les clients ne réapparaissent que vers la fin du processus, car ils sont requis pour les tests d'acceptation et la recette. Ce que les clients découvrent en général à ce moment est que ce qui a été produit n'est

pas exactement ce qu'ils avaient demandé. Pire encore, il est en général trop tard ou trop onéreux de demander des changements, au moins pour la release en préparation. Les clients qui se faisaient une joie de découvrir le résultat sont surpris, frustrés et déçus. Commence alors l'horrible jeu de la recherche du coupable idéal. Le client s'exclame : "Si vous aviez lu mon cahier des charges avec plus de soin, vous auriez produit ce que je voulais vraiment." L'équipe de développement réplique : "Si vous aviez écrit votre document plus précisément, nous aurions produit autre chose. Nous avons produit ce que vous avez demandé !"

Dans Scrum, nous construisons une fonction à la fois, nous ne travaillons pas sur une phase à la fois. Nous terminons toutes les activités relatives à une fonction (conception, codage, intégration et test) pendant un même sprint. Voilà pourquoi le PO doit absolument maintenir un fort niveau d'engagement en permanence. Une interaction forte marquée par des itérations limitées dans le temps réduit de beaucoup les risques de voir le PO se déconnecter de l'équipe de développement. Un autre avantage est qu'il n'y a plus de chasses aux sorcières lorsque la méthode Scrum est bien appliquée !

Collaboration avec les parties prenantes

Le Product Owner est le porte-parole unique de toute la communauté que forment ceux qui ont un intérêt moral ou financier à la réussite du projet, c'est-à-dire les parties prenantes internes et externes. Les parties prenantes internes sont les propriétaires de l'entreprise, la direction générale, les managers de programme, le marketing et les ventes. Les parties prenantes externes sont les clients, les utilisateurs, les partenaires, les organismes de régulation, etc. Il faut que le PO travaille intimement avec toute cette communauté pour collecter des informations et produire une vision cohérente qui va guider le développement du produit.

Si le PO commence à être débordé, il lui devient difficile de collaborer simultanément avec l'équipe de développement et avec toutes les parties prenantes au niveau nécessaire. Il peut arriver que la charge de travail dépasse les capacités d'une seule personne. Dans ce cas, le PO peut demander l'aide d'autres personnes pour bien répondre à la charge du rôle. J'aborderai cet aspect plus loin lorsque je présenterai le concept d'équipe de Product Owners.

Caractéristiques/compétences

La Figure 9.5 présente les caractéristiques essentielles du rôle de Product Owner.

Qui peut être nommé Product Owner ?

Dans la plupart des entreprises qui n'utilisent pas encore Scrum, il est peu probable qu'il existe déjà un rôle intitulé "Product Owner". Comment l'entreprise doit-elle choisir qui nommer à ce rôle ?

J'ai indiqué précédemment dans ce chapitre que le PO devait en permanence faire l'interface entre deux entités : il doit regarder en direction des parties prenantes internes et externes et en direction de l'équipe de développement. Le rôle de PO comporte donc de l'autorité et des responsabilités qui appartiennent habituellement à plusieurs rôles classiques différents. Une définition complète du poste de PO regroupe en partie les cinq rôles suivants : product manager, product marketer, chef de projet/project manager (dont je parle plus en détail dans le Chapitre 13), analyste métier et contrôleur qualité/recette.

Les détails du profil de PO dépendent du type de développement et de l'organisation. Le Tableau 9.1 propose des profils appropriés au rôle de PO pour quatre types de développement différents qui sont décrits ci-dessous.

Tableau 9.1 : Profils de Product Owner pour différents types de développement

<i>Type de développement</i>	<i>Profil de candidat Product Owner</i>
Développement interne	Délégué ou client du service de l'entreprise qui va bénéficier de la solution
Développement commercial	Représentant interne des clients et utilisateurs (en général, un product manager, un product marketer ou un project manager)
Développement externalisé	Représentant ou client de l'entreprise qui achète la solution et en tirera les bénéfices
Équipe composant (développement d'architecture)	En général, un technicien capable de bien trouver des priorités dans le Backlog des éléments techniques

Développements internes

Dans le cas d'un développement interne, le PO devrait être choisi parmi les personnes ayant autorité et issues du groupe qui va bénéficier du produit. Si par exemple c'est un groupe informatique interne qui va produire un système pour le marketing, le PO devra être une personne de l'équipe marketing dotée de pouvoirs suffisants (voir Figure 9.7).